Optimisation of multi-omic genome-scale models: methodologies, hands-on tutorial and perspectives

Supreeta Vijayakumar*, Max Conway*, Pietro Lió and Claudio Angione

(i) Abstract Genome-scale metabolic models are valuable tools for assessing the metabolic potential of living organisms. Being downstream of gene expression, metabolism is being increasingly used as an indicator of the phenotypic outcome for drugs and therapies. We here present a review of the principal methods used for constraint-based modelling in systems biology, and explore how the integration of multi-omic data can be used to improve phenotypic predictions of genome-scale metabolic models. We believe that the large-scale comparison of the metabolic

Supreeta Vijayakumar

Department of Computer Science and Information Systems, Teesside University, UK e-mail: s.vijayakumar@tees.ac.uk

Max Conway

Computer Laboratory, University of Cambridge, UK e-mail: max.conway@cl.cam.ac.uk

Pietro Lió

Computer Laboratory, University of Cambridge, UK e-mail: pietro.lio@cl.cam.ac.uk

Claudio Angione

Department of Computer Science and Information Systems, Teesside University, UK e-mail: c.angione@tees.ac.uk

*These authors contributed equally to this work

response of an organism to different environmental conditions will be an important challenge for genome-scale models. Therefore, within the context of multi-omic methods, we describe a tutorial for multi-objective optimisation using the metabolic and transcriptomics adaptation estimator (METRADE), implemented in MATLAB. METRADE uses microarray and codon usage data to model bacterial metabolic response to environmental conditions (e.g. antibiotics, temperatures, heat shock). Finally, we discuss key considerations for the integration of multi-omic networks into metabolic models, towards automatically extracting knowledge from such models.

(ii) Keywords: Multi-omics, metabolic models, flux-balance analysis, machine learning, data integration, multi-objective optimisation.

1. Introduction

Metabolism is the set of biochemical reactions in a cell which maintain its living state. As these reactions are indispensable, it is vital that metabolic networks in all living organisms are as well-characterised as possible. In the higher organisation level of a microbial community, cells can act as either sinks or sources of metabolites in their environment, as they consistently produce or deplete a range of metabolites in the environmental metabolite pool [1]. Being downstream of gene expression, metabolism is being increasingly used as an indicator of the phenotypic outcome for drugs and therapies, as well as for cancer studies [2].

Constraint based reconstruction and analysis (COBRA) techniques are commonly used for modelling reconstructions of metabolic networks at the genome scale. The most widely used method is flux balance analysis (FBA), which has long been used to mathematically express the flow of metabolites through a network of biochemical pathways. FBA uses the assignment of stoichiometric coefficients to represent each

of the metabolites involved in any given reaction [3]. Through these coefficients, mass-balance constraints can be imposed on the system to identify a range of points representing all possible flux distributions, which correspond to the set of feasible phenotypic states. In this solution space, there exists a global optimal value which satisfies a given objective function (usually the maximisation of biomass). For purposes of mass conservation, all fluxes within this system are calculated under the steady state assumption that the total amount of any metabolite being produced must be equal to the total amount of that metabolite consumed, [4] and that the cell can utilise resources optimally in time-invariant and spatially homogeneous extracellular conditions [5, 6]. Linear programming can be used to maximise an objective function indicating the extent to which each reaction contributes to a certain phenotype, under constraints which can be defined by a cell's metabolic potential, stoichiometry and limits of reaction and transport rates [1].

The main advantage of using FBA is that it does not invariably require the definition of kinetic parameters. In fact, fluxes are calculated in a pseudo-steady state using stoichiometric coefficients and mass balances; this affirms its suitability for building mechanistic predictive models from genome-scale metabolic networks [7]. Using the optimal value obtained through FBA, flux variability analysis (FVA) [8] returns the maximum and minimum values for fluxes through each reaction whilst keeping the formation of biomass to a minimum, which can help in calculating the rate of metabolite consumption or production [9].

More detailed analyses can be carried out to provide a deeper insight into certain aspects of metabolic processes. To overcome the limitations of the steady state assumption, dynamic FBA can be carried out by monitoring time dependent changes in the concentration of metabolites and reaction fluxes over time [5]. This involves calculating the conservation of mass for each of the metabolites consumed and

produced in reactions and imposing additional constraints on the rates of flux changes, non-negative metabolite and flux levels and transport fluxes [10].

Several genome-scale metabolic models are readily available in online repositories such as KEGG [11], BIGG [12], BioCyc [13] and SEED [14]. These are prepared by building a genome-scale reconstruction of all metabolic reactions taking place in the organism followed by manual curation, gap-filling and annotation of specific genes, metabolites and pathways with descriptive metadata. Recently, an increasing number of genome-scale signalling and regulatory networks are also being compiled in order to garner a better understanding of the underlying mechanisms of metabolic pathways [5], and approaches to extract pathway cross-talks have been proposed [15].

Parsimonious enzyme usage FBA (pFBA) is a variant of FBA which aims to maximise the stoichiometric efficiency of a metabolic network by identifying a subset of genes which contribute to maximising the growth rate in silico. These include both essential and non-essential genes, as well as those which are enzymatically and metabolically less efficient and those which are completely unable to carry flux in experimental conditions [16].

For a more detailed introduction to constraint-based metabolic models, the interested reader is referred to the following texts: [17, 18]. After reviewing the available methods for optimisation of metabolic networks, we also provide a tutorial for multi-objective optimisation using METRADE. The tutorial illustrates how to predict bacterial multi-response under varying environmental conditions, by computing the trade-off between contrasting metabolic objectives.

Finally, we recognise that systematic fusion of multiple data types into a single, cohesive network is a challenge faced by many modellers, particularly when measuring bacterial response at multiple omic levels. In view of this, we include a critical per-

spective describing key considerations for the integration of multi-omic networks into metabolic models, towards automatically extracting knowledge from such models.

2. Materials

2.1 Multi-target optimisation of multi-omic metabolic networks

Available methods for analysis of metabolic networks and metabolic engineering usually define gene lethality in terms of effect on the growth rate only. In fact, organisms often have multiple objectives to satisfy in addition to the maximisation of biomass. To this end, a number of approaches have been recently proposed to take into account multi-target optimisation of cellular tasks. Unlike single-objective approaches, these allow for simultaneous maximisation or minimisation of two or more properties of interest.

Gene knockout simulation is one of most consistently used methods for determining the essentiality of genes, and has been successfully applied to the design and optimisation of strains for metabolic engineering. However, it has been contended that single gene perturbations can often fail to capture the essentiality of genes or localise gene function owing to genetic redundancy. As a result, when a metabolic function is encoded by two or more genes, the removal of any one of these genes will not result in an altered phenotype, and it may therefore be falsely concluded that they are superfluous [19]. The regulatory on/off minimisation (ROOM) algorithm uses mixed integer linear programming to predict the metabolic state of an organism following knockout [20]. This is achieved by searching for the flux distribution of the perturbed strain that minimises the number of significant flux changes (which may allude to underlying regulatory changes after knockout) whilst satisfying all

stoichiometric, thermodynamic and flux capacity constraints applied during FBA. On the other hand, multiple genetic perturbations carried out concurrently may lead to issues relating to technical and conceptual scaling. Hence, pairwise gene knockouts may be considered better for identifying which deletions have a damaging effect. For instance, a computational approach has been presented for identifying dosage lethality effects (IDLE) in genome scale models of cancer metabolism [21] using synthetic dosage lethality to simulate the pairwise knockout of non-essential enzymes by overexpressing the first enzyme-coding gene but underexpressing the second.

On the whole, performing complete gene knockouts is still likely to present a number of complications such as: (i) the lack of information regarding the effect of removing essential reactions; (ii) increased compression of the flux distribution following the removal of flux values during knockout; (iii) difficulty in optimising fluxes if they are limited to their Boolean definition of having either a lethal or neutral phenotypic effect [22].

To address the problem of the state-space explosion when considering all possible combinations of multiple gene knockouts, evolutionary algorithms have been proposed, both searching in the discrete space of gene knockouts [23] and in the continuous space of gene partial overexpression/underexpression [24]. This enables the consideration of more than one objective function and expands the phenotypic solution space as there are a greater number of feasible optimal points. Multi-objective optimisation can help to resolve trade-offs between conflicting metabolic objectives through simulating a series of optimal, non-dominated vectors in the multi-dimensional objective space. In metabolic engineering, each vector may represent a Boolean gene knockout strategy, or a real-valued partial knockdown/overexpression strategy. For such vectors, there is no better solution which exists for a given objective without sacrificing the performance of another [25]. This is known as a Pareto front

and enables the consideration of multiple conditions and constraints affecting each objective in a multi-objective optimisation problem.

The key advantage of multi-objective optimisation is that it seeks a trade-off between multiple cellular objectives, without the need to define individual weights and combine them into a single objective [26] or hierarchically order objectives [27]. This eliminates difficulties associated with choosing the most suitable objective function or selecting weights which uniformly represent the Pareto front. The use of multi-objective evolutionary algorithms (MOEAs) such as NSGA-II [28], SPEA2 [29] and MOEA/D [30] quickly renders all Pareto-optimal solutions when objectives are simultaneously optimised. Linear physical programming-based flux balance analysis (LPPFBA) orders objectives by their Pareto-optimal solutions to identify those which are in conflict [31]. This helps to select regions of the solution space which contain feasible fluxes. Optimal flux vectors can be also found using comprehensive polyhedra enumeration flux balance analysis (CoPE-FBA) through finding the topology of sub-networks corresponding to these vectors [32]. In this method, dividing reversible reactions into separate forward and backward reactions further simplifies the solution space for finding non-decomposable flux routes [33].

Multi-objective optimisation can be implemented into FBA using the noninferior set estimation (NISE) method to approximate Pareto curves for conflicting objectives and examine flux at all Pareto-optimal solutions [34]. More recently, variations of MOFBA and MOFVA have been used to compute metabolic trade-offs for multiple species within microbial communities in terms of growth rates and associated reactions [35]. Thermodynamic states have also been incorporated in such analyses to inform responses to environmental conditions. Estimations of maximum yields using single objective optimisation can be extended for multiple objectives to find the area for which one factor cannot be increased without sacrificing another (i.e. a Pareto

surface of yield versus productivity), through which it is possible to devise strategies for improving performance by increasing metabolic flexibility [36].

As a pre-processing step, sensitivity analysis can be carried out to discover the most influential inputs for the multi-objective optimisation problem by interrogating the pathway, reaction or species spaces of the model. In particular, pathway-oriented sensitivity analysis [23] has proved to be useful in metabolic engineering for improving the robustness of strains by determining the most sensitive metabolic pathways; this is achieved by identifying which knockouts or genetic manipulations contribute the most towards a certain output.

2.2 Integration of multi-omic data types into genome-scale metabolic models

Several methods for integration of gene expression data into metabolic models have been proposed; for a comprehensive review, the reader is referred to Machado and Herrgård [37]. However, it has readily been established that multi-omic integration of data allows for a more comprehensive evaluation of model predictions, rather than solely relying on gene expression profiling for the observation of metabolic responses over a range of different environmental conditions. The optimisation of transcriptomic and proteomic layers with respect to different growth conditions serves to refine predictions of metabolic phenotypes (Figure 1).

Regulatory FBA (rFBA) is an extension of FBA which adds the dimension of transcriptional regulation to improve flux predictions for dynamic models by recording transcriptional events and protein activity as well as simulating the uptake of metabolites, biomass production and the secretion of by-products [38]. Alternatively, the probabilistic regulation of metabolism (PROM) method combines gene expression

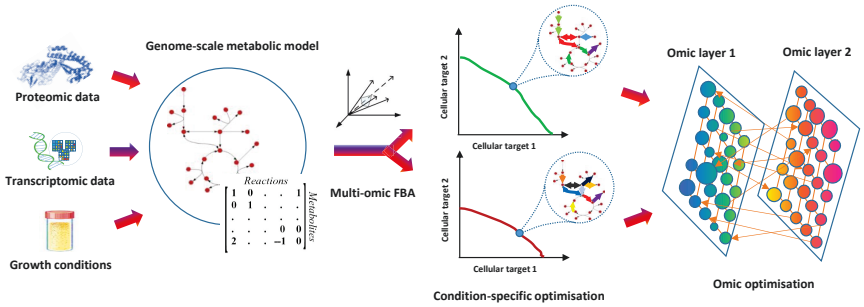

Fig. 1 Through the collection of transcriptomic, proteomic and other omic data across various growth conditions from *in-vivo* experiments and existing literature, a genome-scale metabolic model can be constructed and FBA carried out at multiple levels. The simulation of growth under different conditions allows for condition-specific optimisation of each of the omic layers, which can then be combined to form a multi-omic network.

data with transcriptional regulatory networks by quantifying the interactions from high-throughput data in an automated fashion [39] to overcome limitations associated with Boolean logic. This is achieved through the use of conditional probabilities to represent gene states and gene-transcription factor interactions [40]. Therefore, a greater number of interactions can be modelled, consequently improving the prediction of phenotypic states for various transcriptional perturbations.

Conditional FBA applies conditional dependencies present in the metabolic model as constraints for each flux. In other words, each flux is constrained by the activity of the compound that facilitates it. For example, temporal variations in response to varying light intensity and associated conditional dependencies were included in a constrained genome-scale metabolic model, in order to simulate the phototrophic growth of the cyanobacterium *Synechocystis* sp. PCC 6803 over a diurnal cycle [41]. More recently, a system was devised using *Synechococcus elongatus* PCC 7942 as a model to study issues concerning resource allocation encountered during phototrophic growth [42].

A unified measure of bacterial responses computed by a condition-specific model allows for the detection of coordinated responses shared between different data

types as well as the variation in responses across differing growth conditions. In this regard, a method for the concatenation of disparate omics data types (layers) has been proposed over varying growth conditions (nodes) into an aggregated model [43]. Using multilayer network models, the omics were weighted for the reliability of the flux rate predictions. Additionally, calculating flux distributions with multiple levels allowed for exploration of the total metabolic potential of the organism and the use of a non-binary measure of gene expression. By coupling fluxomic and proteomic data, a novel biological relationship was uncovered between protein structure and translational pausing, as well as an improved in vivo estimation of genome-wide enzyme turnover rates [44]. This approach helped to develop a parameterised model to predict responses to conditions, and consequently inform metabolic cost-benefit ratios at the cellular level.

The minimisation of metabolic adjustments (MOMA) uses quadratic programming to solve its optimisation problem. The objective function is calculated as the distance between two different flux distributions: the flux distribution for optimal growth rate and the flux distribution following the generation of a knockout mutant through genetic perturbation [45]. This accounts for the fact that knockout mutants are likely to display a lower growth rate than the wild type, therefore their flux distribution is better predicted by the minimal flux response to the knockout rather than by an optimal growth rate [46]. The inactivation of genes imposes additional constraints on the system, arguably leading to a shift towards a more valid and biologically meaningful representation of flux distribution as close as possible to that of the wild type [47, 48]. Similarly, integrative omics metabolic analysis (IOMA) uses a mechanistic model to determine reaction rates by incorporating quantitative proteomic and metabolomic data into the model to deliver more accurate predictions of flux alterations following genetic perturbation [49].

In the context of metabolic engineering, multi-omic integration has been used to find strain-specific differences for the improved selection and design of optimal strains. The goal is threefold: (i) maximising the theoretical yield of a particular metabolic product by comparing high flux reactions between strains using physiological data added to the model; (ii) quantifying differential gene expression using transcriptomic profiles; (iii) analysing gene expression across different conditions, thus characterising the specific metabolic capabilities of individual strains [50]. Gene expression measurements can be obtained from microarray and/or RNA sequencing data from public repositories for integration with metabolic networks. Gene inactivity moderated by metabolism and expression (GIMME) is a switch-based algorithm which can be used to perform discretisation (i.e. binary classification) of gene expression data to reduce the amount of experimental noise, by finding inactive genes in the dataset and re-enabling flux associated with false negative values [51]. Chiefly, the algorithm scores the consistency of gene expression data for a given metabolic objective [52].

Conversely, there are a number of valve-based algorithms such as E-flux [53] and METRADE [24], which treat gene expression data as continuous rather than discrete. Lower and upper bounds are set so that the maximum allowable flux for a reaction is a function of the normalised expression of genes controlling that reaction. The idea is to tightly constrain the maximum and minimum flux when the expression for a gene is low, but relaxing these constraints when the expression is high. Due to the addition of these constraints, performing FBA returns an altered flux distribution, which may consequently alter the corresponding metabolic state or optimal metabolic capacity identified. There is another branch of methods which employ 'pruning' so that only a core set of reactions are retained in the metabolic model. Methods using this approach to integrate models with tissue-specific data include MBA [54], FASTCORE [55] and mCADRE [56].

Since an increasing number of genome-scale transcriptional regulatory networks are now available, methods like PROM [39] should be preferred to examine cellular transcriptional activity, as they do not rely on assigning a Boolean on/off state to each gene. Regulatory elements may also be incorporated into models by performing enrichment analysis of transcription factors for differential control of genes [50], or by merging transcriptional regulatory networks with constraint-based metabolic models [57]. A multilayer model was constructed for *Escherichia coli* [58] which merged sub-models of transcriptional regulatory networks, signal transduction pathways and metabolic networks; trained parameters were fed into the model to return information for an objective function and set of constraints with subsequent model predictions improved through supplementation with experimental data. To bridge the gap (and the still debated assumption of strong correlation) between gene expression levels and protein abundance, a method was recently proposed to account for the synonymous codon usage bias [59].

We believe that the large-scale comparison of the metabolic responses between different environmental conditions will be an important challenge for genome-scale modelling. In the following section, a tutorial is presented for METRADE [24], which gives a step-by-step guide to perform optimisation of metabolic models. This is achieved by mapping gene expression values to the objective space of a genome-scale metabolic model and performing multi-objective optimisation for identifying optimal phenotypes through the comparison of predicted flux rates for multiple objectives. METRADE develops a multi-omic model of *Escherichia coli* that includes a multi-objective optimisation algorithm to find the allowable and optimal metabolic phenotypes through concurrent maximisation or minimisation of multiple metabolic markers. A number of experimental conditions are mapped to the model through transcriptomic data, and then mapped to a phenotypic multidimensional objective space.

3. Methods

The framework for the metabolic and transcriptomics adaptation estimator (METRADE) incorporates multi-objective optimisation by constructing a Pareto front which displays gene expression profiles and codon usage arrays in a condition-phase space, where each profile is associated with a growth condition [24]. This allows for comparison of objectives to identify the best trade-off, where the maximal number of cellular objectives are simultaneously optimised. Sets of Pareto-optimal solutions in the front may be represented using a hypervolume indicator [60], enabling comparison between mapped conditions and examination of Pareto set evolution towards an optimal configuration over time.

In the context of metabolic engineering, strains may be compared for their ability to simultaneously fulfil multiple objectives and optimise production of multiple metabolites at the same time. It is also possible to establish the optimal growth conditions necessary to achieve this output and devise strategies for further optimisation through performing gene knockouts or changing flux rates *in-vitro*. Additional insights into bacterial adaptability can be obtained through principal component analysis (PCA) [61], pseudospectra [62], and community detection [63]. PCA aids investigation of components (i.e. expression profiles) with the greatest variance for multiple objectives, whereas the pseudospectra and community detection methods elucidate the community structure of bacteria in the condition phase-space.

METRADE can be run (i) as a standalone program to find the optimal gene expression values for maximisation of given cellular objectives, and (ii) on a dataset of growth conditions to find the predicted flux rates in any given condition.

3.1 Initial settings

METRADE is fully compatible with the COBRA 2.0 toolbox [64]. The full code needed for METRADE can be downloaded from http://www.nature.com/articles/srep15147. The user can download COBRA toolbox for MATLAB from http://opencobra.github.io/ and set the local COBRA folder in the MATLAB path with the instruction

```
addpath(genpath('local_path_to_COBRA_toolbox'));
```

Load the model e.g. the one included in the folder, *Escherichia coli* iJO1366 [65] with acetate-biomass set as objectives:

```
load('iJO1366_Ecoli_ac.mat')
```

The variable *fbamodel.f* selects the first objective (default: biomass). The variable *fbamodel.g* selects the second objective (default: acetate). To find the indices of the reactions for oxygen, succinate and acetate import/export, type

```
ix_o2 = find(ismember(fbamodel.rxns, 'EX_o2(e)')==1);
ix_succ = find(ismember(fbamodel.rxns, 'EX_succ(e)')==1);
ix_ac = find(ismember(fbamodel.rxns, 'EX_ac(e)')==1);
```

The pair of objective functions can be changed. For instance, to change the second objective to succinate, use:

```
fbamodel.g(ix_ac) = 0;
fbamodel.g(ix_succ) = 1;
```

To change between aerobic and anaerobic conditions, we have to set a new lower bound for the reaction importing oxygen. For instance, to simulate an anaerobic condition, set the lower bound to zero (no import allowed).

```
fbamodel.lb(ix_o2) = 0;
```

Note that default aerobic conditions are considered with a lower bound of -10 mmol/h/gDW. A negative lower bound represents the maximum rate available for the import of that metabolite. Anaerobic conditions are with a null lower bound.

3.2 Mapping gene expression compendia to multidimensional objective spaces

Run *pareto_microarray_fluxes.m*. This will generate flux rates for 466 given growth conditions [66], and will save them in a variable called *fluxes*. The two fluxes chosen as objectives (default: biomass and acetate) will be saved in a file called *points*. These values represent the coordinates of the points in the bi-dimensional objective space shown in the paper.

Listing 1 Mapping growth conditions on multidimensional phenotypic spaces

```
format long

% starts the parallel toolbox to use four cores
if (matlabpool('size') == 0)  %opens only if it is closed
    matlabpool('open','local',4)
end

% initialises the Cobra toolbox
initCobraToolbox

```

```
% loads variables
load('genes.mat');
load('reaction_expression.mat');
load('probe_genes.mat');
load('glucose.mat');
load('oxygen.mat');
load('name_conditions_with_replicates.mat');
load('name_conditions.mat');
if evalin('base','exist(''data_only'',''var'')')==0
    load('data_only.mat');
end
load('gene_variances.mat')
max_gene_importance = 10000;


%The following instructions find the locations of the gene 'bXXXX
    ' (genes in the fbamodel) in the array probe_genes (sequence
    of genes appearing in the microarray data available)

position_gene = cell(length(genes),1);

for i=1:length(genes)
    matches = strfind(probe_genes,genes{i});
    position_gene{i} = find(~cellfun('isempty', matches));
end

```

```
35 points = zeros(size(data_only,2),2);     %table of points
       coordinates. The number of points is equal to the number of
       conditions in the microarray data
36 gene_importance = zeros(length(genes),1);    %array of the
       coefficients indicating gene importance. The size is equal to
        the number of genes in the model
37
38 min_var = min(gene_variances);
39 probe_gene_importance = max_gene_importance * 1./(gene_variances/
       min_var);  %this way the importance of a gene can range from
        0 to max_gene_importance
40
41 for i=1:length(genes)
42     if isempty(position_gene{i})
43         gene_importance(i) = max_gene_importance/2;
44     else
45         gene_importance(i) = probe_gene_importance(position_gene{
               i});
46     end
47
48 end
49
50 number_conditions = size(data_only,2);
51 points = zeros(number_conditions,2);
52 fluxes = zeros (length(reaction_expression),number_conditions);
53
54 parfor_progress(number_conditions); % Initialize
```

```
55
56 for index_cond = 1 : number_conditions
57     microarray_data = data_only(:,index_cond);
58     [v1, out] =  process_conditions(microarray_data, index_cond,
           genes, position_gene, fbamodel, oxygen, glucose,
           name_conditions, name_conditions_with_replicates,
           reaction_expression, gene_importance);  %it is necessary
           to pass the original fbamodel that will be changed in the
            subfunction (oxygen will be put to zero or not according
            to the anaerobic or aerobic condition, the default
           condition in fbamodel is aerobic)
59     points(index_cond,:) = out;
60     fluxes(:,index_cond) = v1;
61     disp(index_cond);
62     parfor_progress;
63 end
64
65
66 string = ['points_gene_importance_' num2str(max_gene_importance)
       ];
67 save(string, 'points');
68
69 string = ['fluxes_gene_importance_' num2str(max_gene_importance)
       ];
70 save(string, 'fluxes');
```

3.3 Multi-objective optimisation of gene expression

We will now solve the inverse problem, namely finding the best genome-wide expression values that allow for the maximisation of two given cellular objectives. This part implements a multi-objective optimisation algorithm using a genetic algorithm based on NSGA-II [28] (the comments in the genetic algorithm code below are adapted from the original NSGA-II implementation). The trade-off between multiple metabolic objectives is found as a result. Such methods can guide genetic engineering to find the best gene expression values for specific goals. Furthermore, they can elucidate the metabolic capability of an organism and the relationship between contrasting cellular objectives.

To start the optimisation, launch *RUN.m* (by editing the file, it is possible to set the number of cores and select the model). The number of populations of the optimisation algorithm is set to 150 by default, and the number of individuals per population is set to 100. We suggest keeping this proportion. The results in the METRADE paper have been obtained with 1500 populations of 1000 individuals each.

Listing 2 Multi-objective optimisation of metabolic models. The code has been parallelised to work on all the available cores when executed on a multi-core processor.

```
1
2  load 'genes.mat';
3  load 'reaction_expression.mat';
4
5  M = 2; %number of objective functions
6  V = length(genes); %length of the input individuals without
       ranking, crowding distance and outputs
7  N = pop; %population size
8
```

```
min_range = zeros(1,V);     %the expression of each gene is >= 0
max_range = 100*ones(1,V);  %the expression of each gene is <= 100


%% Initialise the population

if ((last_gen==0))
    chromosome=ones(pop,V+M);   %gene expressions initialized as 1, i.e. all the genes are normally expressed (reference state). A chromosome means, in our case, an array of gene expression values

%    chromosome = chromosome + 0.1.*(rand(pop,V+M)-0.5.*ones(pop, V+M));  %adds some initial random noise
     chromosome = chromosome + 2.*(rand(pop,V+M)-0.5.*ones(pop,V+M));  %adds some initial random noise
    [v1, fmax] = flux_balance(fbamodel,true);

    for i=1:pop
        chromosome(i,V+1)= - fmax;% acetate
        chromosome(i,V+2)= - fbamodel.f' * v1;%biomass
    end
    %% Sort the initialised population
    % Sort the population using non-domination-sort. This returns two columns for each individual which are the rank and the crowding distance corresponding to their position in
```

```
        the front they belong. At this stage the rank and the
        crowding distance for each chromosome is added to the
        chromosome vector for easy of computation.
    chromosome = non_domination_sort_mod(chromosome, M, V);
else
    sol=['solution' num2str(last_gen) '.mat'];
    load(sol);
end

%% Start the evolution process
% The following are performed in each generation
% * Select the parents which are fit for reproduction
% * Perform crossover and mutation operators on the selected
    parents
% * Perform selection from the parents and the offspring
% * Replace the unfit individuals with the fit individuals to
    maintain a
%   constant population size.

for i = last_gen+1 : gen
    % Select the parents
    % Parents are selected for reproduction to generate offspring
        . The original NSGA-II uses a binary tournament selection
         based on the crowded-comparison operator. The arguments
        are
    % pool - size of the mating pool. It is common to have this
        to be half the population size.
```

```
46  % tour - Tournament size. Original NSGA-II uses a binary
        tournament selection, but to see the effect of tournament
         size this is kept arbitrary, to be chosen by the user.
47
48  pool = round(pop/2);
49  tour = 2;
50  % Selection process
51  % A binary tournament selection is employed in NSGA-II. In a
        binary tournament selection process two individuals are
        selected at random and their fitness is compared. The
        individual with better fitness is selected as a parent.
        Tournament selection is carried out until the pool size
        is filled. Basically a pool size is the number of parents
         to be selected. The input arguments to the function
        tournament_selection are chromosome, pool, tour. The
        function uses only the information from last two elements
         in the chromosome vector.
52  % The last element has the crowding distance information
        while the penultimate element has the rank information.
        Selection is based on rank and if individuals with same
        rank are encountered, crowding distance is compared. A
        lower rank and higher crowding distance is the selection
        criteria.
53  parent_chromosome = tournament_selection(chromosome, pool,
        tour);
54
55  % We now apply crossover and mutation operators
```

```
56      mu = 20;
57      mum = 20;
58      if (num_cores>1)
59          offspring_chromosome = genetic_operator_parallel(
                parent_chromosome, M, V, mu, mum, min_range,
                max_range, fbamodel, genes, reaction_expression);
60      else
61          offspring_chromosome = genetic_operator(parent_chromosome
                , M, V, mu, mum, min_range, max_range, fbamodel,
                genes, reaction_expression);
62      end
63
64
65      % We now create the intermediate population, namely the
            combined population of parents and offspring of the
            current generation. The population size is two times the
            initial population.
66
67      [main_pop,temp] = size(chromosome);
68      [offspring_pop,temp] = size(offspring_chromosome);
69      clear temp
70      % intermediate_chromosome is a concatenation of current
            population and the offspring population.
71      intermediate_chromosome(1:main_pop,:) = chromosome;
72      intermediate_chromosome(main_pop + 1 : main_pop +
            offspring_pop,1 : M+V) = offspring_chromosome;
73
```

```
74      % Non-domination-sort of intermediate population
75      % The intermediate population is sorted again based on non-
            domination sort before the replacement operator is
            performed on the intermediate population.
76      intermediate_chromosome = non_domination_sort_mod(
            intermediate_chromosome, M, V);
77
78      % Perform Selection
79      % Once the intermediate population is sorted only the best
            solution is selected based on it rank and crowding
            distance. Each front is filled in ascending order until
            the addition of population size is reached. The last
            front is included in the population based on the
            individuals with least crowding distance
80      chromosome = replace_chromosome(intermediate_chromosome, M, V
            , pop);
81  %   chromosome = delete_redundant(chromosome,fbamodel);
82      solution=['solution' num2str(i)];
83      save(solution, 'chromosome');
84
85  end
```

After the optimisation, *append_and_plot_solutions.m* computes the Pareto front. The file *non_dominated.mat* contains all the Pareto optimal points, while *others.mat* contains the dominated points. The first two columns of both output files contain the predicted values for the two objective functions. The 4th column is the number

of population in which that solution has been found, while the 5th column is the position of that solution in that population.

Finally, *plot_and_export_color.m* plots the final version of the Pareto front. An example of a Pareto front obtained for 12propanediol and biomass is shown in Figure 2.

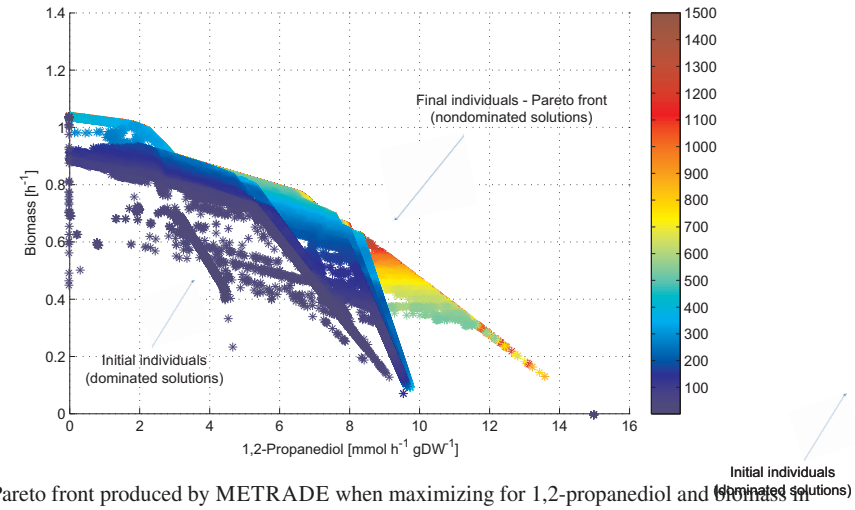

Fig. 2 Pareto front produced by METRADE when maximizing for 1,2-propanediol and biomass in *E. coli* (adapted from [24]). The trade-off sheds light on the regions where the bacterium operates. Solutions are asterisks denoted by progressively warmer colours according to the time step of the genetic algorithm in which they have been generated. Although discrete, the Pareto front can be approximated by a piecewise linear function.

To validate with the proteomic dataset by Hui et al. [67] included in METRADE, load *iJO1366_Ecoli_ac_lactoseMedium.mat* and run *pareto_proteomic.m*. The dataset is composed of 14 expression profiles in different growth conditions with: (i) titrated catabolic flux through controlled inducible expression of the lacY gene; (ii) titrated anabolic flux through controlled expression of GOGAT; (iii) inhibition of protein synthesis with chloramphenicol. To run the pseudospectrum analysis on the growth conditions as detailed in METRADE, run *plot_eigenvector.m*. The code requires an updated *eigtoollib* toolbox.

There are numerous factors to consider when integrating such multi-omic datasets into metabolic models, many of which are discussed in the following perspective. In order to extract the most meaning from multi-omic models, systematic fusion of the multiple data types into a single, cohesive network is essential for measuring bacterial response at multiple omic levels. Whilst considering the structure of multi-omic data to be used for integration, the techniques used to integrate these data into the model are of equal importance.

4. Notes

4.1 Omic network integration in metabolic models: a (critical) perspective

A large proportion of the techniques which incorporate multi-omic methods into metabolic modelling involve using other omics to constrain the metabolome: they are one-way procedures. However, to properly interpret the results of these procedures, techniques are required which can integrate the different datasets to produce something that is easier to interpret than the separate datasets, and then provide feedback on how those separate datasets affected the integrated dataset. For example, in gene expression constrained FBA methods, the genome and metabolome are integrated into a combined model that enumerates feasible metabolic states. However, the resulting model is inherently complex: the actual relationship between a particular gene and a particular outcome can be hard to understand, even though it is deterministic in the model. Additionally, there is a lack of consensus about the best approach to take when estimating flux rates in different conditions.

Any approach based upon FBA has an inherently linear character: the outputs (fluxes of interest) are linearly dependent on some subset of the inputs (the bounds and objective function). The complexity comes from the fact that, while the output is only linearly dependent on a small fraction of the inputs in any given configuration, all of the other inputs affect which subset this is. This relationship is a piecewise linear equation with a large number of terms, but where most of the coefficients are zero in any given piece. This means that the challenge in understanding these models in an intuitive way is not so much in understanding how each variable affects the model, as when.

When looking to understand the effects of genetic or proteomic data on simulated phenotypes, naturally, the first place to start is at techniques used for understanding the effects of genetic or proteomic data on real phenotypes. With regression style techniques, it becomes clear that the reaction rates induced by FBA are often multi-modal, since the best values are likely to be at either the maximum or minimum of the possible range. This multimodality violates normality assumptions, and it is therefore difficult to sensibly normalise such distributions. This issue has been demonstrated in a correlation analysis between expression levels and Pareto front position [22]. More specifically, even if there are several layers of normalisation and the Pareto front acts to smooth flux values, there are two clear peaks in the distribution of flux rates. Figure 3 shows how this pattern occurs across a number of reactions in a knockout simulation.

The obvious choice when faced with distributions with several narrow peaks is regard these values as fully categorical. However, this approach eventually ends up mired in overfitting; a good approach to combat this is to incorporate structure from the network, e.g. by using a network regularised regression [68] technique to tie the values at nodes to those at nearby nodes.

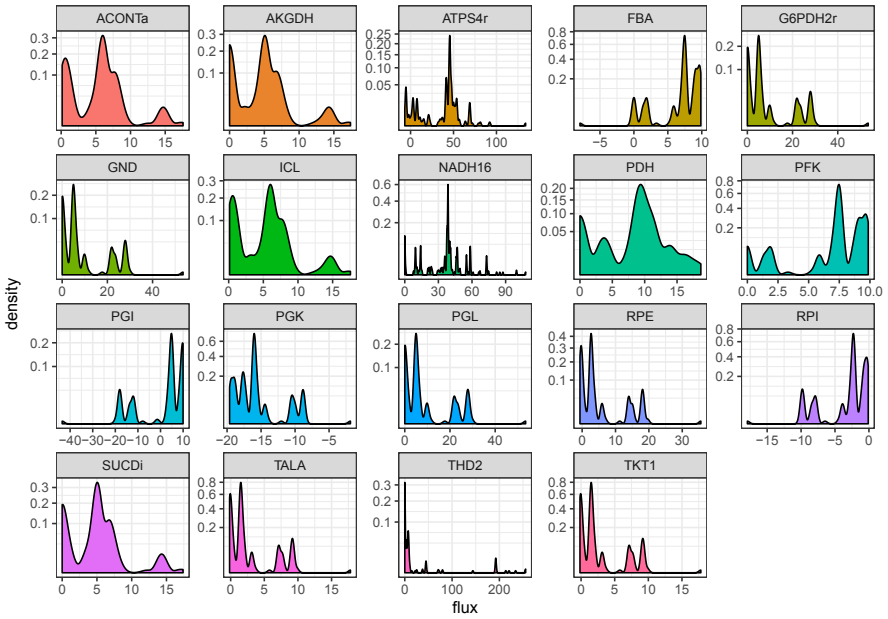

Fig. 3 Density plots of reaction fluxes for 19 reactions across 4560 simulations of one and two reaction knockouts on a model of *E. coli* core metabolism. Data was filtered to remove fluxes for reactions when they were knocked out, to remove simulations with low biomass flux, and to remove reactions with low variation. These reactions all show unsurprising peaks at a flux of 0, but more interestingly show a multimodal distribution, with a small number of other preferred values.

Using multi-omic data, it is possible to go a step further than network regularised regression, and merge multiple omic layers together to form a single network where the value at each node incorporates information both from equivalent nodes in multiple layers, and also neighbours at each level. For instance, Similarity Network Fusion has been proposed to integrate information from a large number of simulations in genotype, metabolome and phenotype domains [43]. This step was as an unsupervised precursor to a supervised decision tree algorithm, which was used to explore the information that various reactions supply about phenotypes.

Ultimately, however, these techniques can only go so far. At their best, they identify under what circumstances certain variables are important, what their effects are, and how they can be clustered. This is a good start, but in order to understand why

variables have the effects they do, a view on the network is required that is simple enough to understand but contains the detail necessary to elucidate a given type of regulation. It is not clear at this stage whether it is better to approach this through general statistical learning techniques or more domain-specific analytical techniques. Either way, it appears to be a goal that will be widely useful for the systems biology community.

5. References

[1] Louca S, Doebeli M (2015) Calibration and analysis of genome-based models for microbial ecology. Elife 4:e08,208

[2] Nilsson A, Nielsen J (2016) Genome scale metabolic modeling of cancer. Metabolic Engineering

[3] Orth JD, Thiele I, Palsson BØ (2010) What is flux balance analysis? Nature Biotechnology 28(3):245–248

[4] Zieliński ŁP, Smith AC, Smith AG, Robinson AJ (2016) Metabolic flexibility of mitochondrial respiratory chain disorders predicted by computer modelling. Mitochondrion 31:45–55

[5] Palsson BØ (2011) Systems biology: simulation of dynamic network states. Cambridge University Press

[6] Jayaraman A, Hahn J (2009) Methods in Bioengineering: Systems Analysis of Biological Networks. Artech House methods in bioengineering series, Artech House, URL https://books.google.co.uk/books?id=Haod3KR-tR8C

[7] Bordbar A, Monk JM, King ZA, Palsson BO (2014) Constraint-based models predict metabolic and associated cellular functions. Nature Reviews Genetics

15(2):107–120

[8] Burgard AP, Vaidyaraman S, Maranas CD (2001) Minimal reaction sets for escherichia coli metabolism under different growth requirements and uptake environments. Biotechnology progress 17(5):791–797

[9] Mahadevan R, Schilling C (2003) The effects of alternate optimal solutions in constraint-based genome-scale metabolic models. Metabolic engineering 5(4):264–276

[10] Mahadevan R, Edwards JS, Doyle FJ (2002) Dynamic flux balance analysis of diauxic growth in escherichia coli. Biophysical journal 83(3):1331–1340

[11] Kanehisa M, Furumichi M, Tanabe M, Sato Y, Morishima K (2016) Kegg: new perspectives on genomes, pathways, diseases and drugs. Nucleic Acids Research p gkw1092

[12] King ZA, Lu J, Dräger A, Miller P, Federowicz S, Lerman JA, Ebrahim A, Palsson BO, Lewis NE (2016) Bigg models: A platform for integrating, standardizing and sharing genome-scale models. Nucleic acids research 44(D1):D515–D522

[13] Caspi R, Billington R, Ferrer L, Foerster H, Fulcher CA, Keseler IM, Kothari A, Krummenacker M, Latendresse M, Mueller LA, et al (2016) The metacyc database of metabolic pathways and enzymes and the biocyc collection of pathway/genome databases. Nucleic acids research 44(D1):D471–D480

[14] Devoid S, Overbeek R, DeJongh M, Vonstein V, Best AA, Henry C (2013) Automated genome annotation and metabolic model reconstruction in the seed and model seed. Systems Metabolic Engineering: Methods and Protocols pp 17–45

[15] Angione C, Pratanwanich N, Lió P (2015) A hybrid of metabolic flux analysis and bayesian factor modeling for multiomic temporal pathway activation. ACS synthetic biology 4(8):880–889

[16] Lewis NE, Hixson KK, Conrad TM, Lerman JA, Charusanti P, Polpitiya AD, Adkins JN, Schramm G, Purvine SO, Lopez-Ferrer D, et al (2010) Omic data from evolved e. coli are consistent with computed optimal growth from genome-scale models. Molecular systems biology 6(1):390

[17] Palsson B (2015) Systems Biology: Constraint-based Reconstruction and Analysis. Cambridge University Press, URL https://books.google.co.uk/books?id=QNBpBgAAQBAJ

[18] Voigt C (2011) Synthetic Biology, Part B: Computer Aided Design and DNA Assembly. Methods in Enzymology, Elsevier Science, URL https://books.google.co.uk/books?id=9uPvZWiabr4C

[19] Deutscher D, Meilijson I, Schuster S, Ruppin E (2008) Can single knockouts accurately single out gene functions? BMC Systems Biology 2(1):50

[20] Shlomi T, Berkman O, Ruppin E (2005) Regulatory on/off minimization of metabolic flux changes after genetic perturbations. Proceedings of the National Academy of Sciences of the United States of America 102(21):7695–7700

[21] Megchelenbrink W, Katzir R, Lu X, Ruppin E, Notebaart RA (2015) Synthetic dosage lethality in the human metabolic network is highly predictive of tumor growth and cancer patient survival. Proceedings of the National Academy of Sciences 112(39):12,217–12,222

[22] Conway M, Angione C, Liò P (2016) Iterative multi level calibration of metabolic networks. Current Bioinformatics 11(1):93–105

[23] Costanza J, Carapezza G, Angione C, Lió P, Nicosia G (2012) Robust design of microbial strains. Bioinformatics 28(23):3097–3104

[24] Angione C, Lió P (2015) Predictive analytics of environmental adaptability in multi-omic network models. Scientific Reports 5:15,147

[25] Angione C, Costanza J, Carapezza G, Lió P, Nicosia G (2015) Multi-target analysis and design of mitochondrial metabolism. PloS one 10(9):e0133,825

[26] Xu G (2011) An iterative strategy for bi-objective optimization of metabolic pathways. In: 2011 Fourth International Joint Conference on Computational Sciences and Optimization

[27] Sendin J, Exler O, Banga JR (2010) Multi-objective mixed integer strategy for the optimisation of biological networks. IET systems biology 4(3):236–248

[28] Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multi-objective genetic algorithm: Nsga-ii. Trans Evol Comp 6(2):182–197, DOI 10.1109/4235.996017, URL http://dx.doi.org/10.1109/4235.996017

[29] Zitzler E, Laumanns M, Thiele L, et al (2001) Spea2: Improving the strength pareto evolutionary algorithm

[30] Zhang Q, Li H (2007) Moea/d: A multiobjective evolutionary algorithm based on decomposition. IEEE Transactions on evolutionary computation 11(6):712–731

[31] Nagrath D, Avila-Elchiver M, Berthiaume F, Tilles AW, Messac A, Yarmush ML (2010) Soft constraints-based multiobjective framework for flux balance analysis. Metabolic engineering 12(5):429–445

[32] Kelk SM, Olivier BG, Stougie L, Bruggeman FJ (2012) Optimal flux spaces of genome-scale stoichiometric models are determined by a few subnetworks. Scientific reports 2:580

[33] Maarleveld TR, Wortel MT, Olivier BG, Teusink B, Bruggeman FJ (2015) Interplay between constraints, objectives, and optimality for genome-scale stoichiometric models. PLoS Comput Biol 11(4):e1004,166

[34] Oh YG, Lee DY, Lee SY, Park S (2009) Multiobjective flux balancing using the nise method for metabolic network analysis. Biotechnology progress 25(4):999–1008

[35] Budinich M, Bourdon J, Larhlimi A, Eveillard D (2017) A multi-objective constraint-based approach for modeling genome-scale microbial ecosystems. PloS one 12(2):e0171,744

[36] John PCS, Crowley MF, Bomble YJ (2016) Efficient estimation of the maximum metabolic productivity of batch systems. arXiv preprint arXiv:161001114

[37] Machado D, Herrgård M (2014) Systematic evaluation of methods for integration of transcriptomic data into constraint-based models of metabolism. PLoS Comput Biol 10(4):e1003,580

[38] Covert MW, Schilling CH, Palsson B (2001) Regulation of gene expression in flux balance models of metabolism. Journal of theoretical biology 213(1):73–88

[39] Chandrasekaran S, Price ND (2010) Probabilistic integrative modeling of genome-scale metabolic and regulatory networks in escherichia coli and mycobacterium tuberculosis. Proceedings of the National Academy of Sciences 107(41):17,845–17,850

[40] Chandrasekaran S, Price ND (2013) Metabolic constraint-based refinement of transcriptional regulatory networks. PLoS Comput Biol 9(12):e1003,370

[41] Rügen M, Bockmayr A, Steuer R (2015) Elucidating temporal resource allocation and diurnal dynamics in phototrophic metabolism using conditional fba. Scientific Reports 5

[42] Reimers AM, Knoop H, Bockmayr A, Steuer R (2016) Evaluating the stoichiometric and energetic constraints of cyanobacterial diurnal growth. arXiv preprint arXiv:161006859

[43] Angione C, Conway M, Lió P (2016) Multiplex methods provide effective integration of multi-omic data in genome-scale models. BMC bioinformatics 17(4):83

[44] Ebrahim A, Brunk E, Tan J, O'Brien EJ, Kim D, Szubin R, Lerman JA, Lechner A, Sastry A, Bordbar A, et al (2016) Multi-omic data integration enables discovery of hidden biological regularities. Nature Communications 7

[45] Segre D, Vitkup D, Church GM (2002) Analysis of optimality in natural and perturbed metabolic networks. Proceedings of the National Academy of Sciences 99(23):15,112–15,117

[46] Raval A, Ray A (2013) Introduction to biological networks. CRC Press

[47] Machado D, Costa RS, Ferreira EC, Rocha I, Tidor B (2012) Exploring the gap between dynamic and constraint-based models of metabolism. Metabolic engineering 14(2):112–119

[48] Brochado AR, Andrejev S, Maranas CD, Patil KR (2012) Impact of stoichiometry representation on simulation of genotype-phenotype relationships in metabolic networks. PLoS Comput Biol 8(11):e1002,758

[49] Yizhak K, Benyamini T, Liebermeister W, Ruppin E, Shlomi T (2010) Integrating quantitative proteomics and metabolomics with a genome-scale metabolic network model. Bioinformatics 26(12):i255–i260

[50] Monk JM, Koza A, Campodonico MA, Machado D, Seoane JM, Palsson BO, Herrgård MJ, Feist AM (2016) Multi-omics quantification of species variation

of escherichia coli links molecular features with strain phenotypes. Cell Systems 3(3):238–251

[51] Vivek-Ananth R, Samal A (2016) Advances in the integration of transcriptional regulatory information into genome-scale metabolic models. Biosystems

[52] Becker SA, Palsson BØ (2008) Context-specific metabolic networks are consistent with experiments. PLoS Comput Biol 4(5):e1000,082

[53] Colijn C, Brandes A, Zucker J, Lun DS, Weiner B, Farhat MR, Cheng TY, Moody DB, Murray M, Galagan JE (2009) Interpreting expression data with metabolic flux models: predicting mycobacterium tuberculosis mycolic acid production. PLoS Comput Biol 5(8):e1000,489

[54] Jerby L, Shlomi T, Ruppin E (2010) Computational reconstruction of tissue-specific metabolic models: application to human liver metabolism. Molecular systems biology 6(1):401

[55] Vlassis N, Pacheco MP, Sauter T (2014) Fast reconstruction of compact context-specific metabolic network models. PLoS Comput Biol 10(1):e1003,424

[56] Wang Y, Eddy JA, Price ND (2012) Reconstruction of genome-scale metabolic models for 126 human tissues using mcadre. BMC systems biology 6(1):153

[57] Imam S, Schäuble S, Brooks AN, Baliga NS, Price ND (2015) Data-driven integration of genome-scale regulatory and metabolic network models. Frontiers in microbiology 6:409

[58] Carrera J, Estrela R, Luo J, Rai N, Tsoukalas A, Tagkopoulos I (2014) An integrative, multi-scale, genome-wide model reveals the phenotypic landscape of escherichia coli. Molecular systems biology 10(7):735

[59] Kashaf SS, Angione C, Lió P (2017) Making life difficult for clostridium difficile: augmenting the pathogen's metabolic model with transcriptomic and

codon usage data for better therapeutic target characterization. BMC Systems Biology 11(1):25

[60] Zitzler E, Brockhoff D, Thiele L (2007) The hypervolume indicator revisited: On the design of pareto-compliant indicators via weighted integration. In: International Conference on Evolutionary Multi-Criterion Optimization, Springer, pp 862–876

[61] Ringnér M (2008) What is principal component analysis? Nature biotechnology 26(3):303

[62] Trefethen LN, Embree M (2005) Spectra and pseudospectra: the behavior of nonnormal matrices and operators. Princeton University Press

[63] Newman M (2013) Spectral community detection in sparse networks. arXiv preprint arXiv:13086494

[64] Schellenberger J, Que R, Fleming RM, Thiele I, Orth JD, Feist AM, Zielinski DC, Bordbar A, Lewis NE, Rahmanian S, et al (2011) Quantitative prediction of cellular metabolism with constraint-based models: the cobra toolbox v2. 0. Nature protocols 6(9):1290–1307

[65] Orth J, Conrad T, Na J, Lerman J, Nam H, Feist A, Palsson B (2011) A comprehensive genome-scale reconstruction of escherichia coli metabolism. Molecular systems biology 7(1):535

[66] Faith JJ, Hayete B, Thaden JT, Mogno I, Wierzbowski J, Cottarel G, Kasif S, Collins JJ, Gardner TS (2007) Large-scale mapping and validation of escherichia coli transcriptional regulation from a compendium of expression profiles. PLoS biol 5(1):e8

[67] Hui S, Silverman JM, Chen SS, Erickson DW, Basan M, Wang J, Hwa T, Williamson JR (2015) Quantitative proteomic analysis reveals a simple strategy

of global resource allocation in bacteria. Molecular systems biology 11(2):784

[68] Li C, Li H (2008) Network-constrained regularization and variable selection for analysis of genomic data. Bioinformatics 24(9):1175–1182